

In the claims:

For the Examiner's convenience, all pending claims are presented below with changes shown in accordance with the mandatory amendment format.

1. (Original) A method comprising:
identifying a predefined behavior of a virtual machine monitor (VMM) with respect to one or more virtual machines (VMs); and
utilizing processor-managed resources associated with the one or more VMs based on the predefined behavior of the VMM.
2. (Original) The method of claim 1 wherein the predefined behavior of the VMM is any one of a first-time invocation of a VM, a subsequent invocation of a VM, a last invocation of a VM, and a modification of content of a virtual machine control structure (VMCS) associated with a VM.
3. (Original) The method of claim 1 wherein identifying a predefined behavior of a VMM comprises receiving an indication of the predefined behavior from the VMM.
4. (Original) The method of claim 3 wherein the indication is received via an instruction executed by the VMM.
5. (Original) The method of claim 4 wherein the instruction executed by the VMM is any one of a VM launch instruction, a VM resume instruction, a virtual machine control structure (VMCS) access instruction, and a VMCS clear instruction.
6. (Original) The method of claim 1 wherein identifying a predefined behavior of a VMM comprises determining the predefined behavior of the VMM by logic within a processor.

7. (Original) The method of claim 6 wherein the logic within the processor is prediction logic.

8. (Original) The method of claim 1 wherein utilization of processor-managed resources includes at least one of allocation of one or more processor-managed resources, de-allocation of one or more processor-managed resources, verification of data stored in one or more processor-managed resources, invalidation of data stored in one or more processor-managed resources, and loading of data into one or more processor-managed resources.

9. (Previously Presented) A method comprising:

determining that a transition from a virtual machine monitor (VMM) to a virtual machine (VM) is about to occur;

determining a type of the transition, the type of the transition being based on invocation information of the VM; and

notifying a processor of the type of the transition.

10. (Original) The method of claim 9 wherein notifying the processor comprises executing an instruction associated with the type of the transition.

11. (Original) The method of claim 9 wherein the type of the transition is any one of an initial transfer to the VM and a subsequent transfer to the VM.

12. (Original) The method of claim 11 further comprising:

in response to determining that the transition is an initial transfer to the VM, allocating a memory region for a new virtual machine control structure (VMCS) associated with the VM, and requesting the processor to activate the new VMCS.

13. (Original) The method of claim 12 wherein requesting the processor to activate the new VMCS comprises executing a VMCS pointer load instruction including a pointer to the

new VMCS as an operand.

14. (Original) The method of claim 12 further comprising requesting the processor to initialize the new VMCS.

15. (Original) The method of claim 14 wherein requesting the processor to initialize the new VMCS comprises executing a VMCS clear instruction including the pointer to the new VMCS as an operand.

16. (Original) The method of claim 12 further comprising:

upon requesting the processor to activate the new VMCS, requesting the processor to set execution control information, VMM state information and VM state information in the new VMCS.

17. (Original) The method of claim 16 wherein requesting the processor to set execution control information, VMM state information and VM state information in the new VMCS comprises executing a VMCS write instruction having an operand that identifies a component of the new VMCS to which data is to be written.

18. (Original) The method of claim 11 further comprising:

in response to determining that the transition is a subsequent transfer to the VM, requesting the processor to update content of a virtual machine control structure (VMCS).

19. (Previously Presented) A method comprising:

receiving, from a virtual machine monitor (VMM), a request to perform a transition from the VMM to a virtual machine (VM), the request indicating a type of the transition, the type of the transition being based on invocation information of the VM; and performing a set of operations according to the type of the transition.

20. (Original) The method of claim 19 wherein receiving the request to perform the

transition comprises identifying a VMM execution of an instruction associated with the type of the transition.

21. (Original) The method of claim 19 wherein the type of the transition is any one of an initial transfer to the VM and a subsequent transfer to the VM.

22. (Original) The method of claim 19 wherein further comprising:
prior to receiving the request to perform the transition, receiving from the VMM a pointer to a virtual machine control structure (VMCS) associated with the VM.

23. (Original) The method of claim 22 wherein the pointer to the VMCS is included as an operand of a VMCS pointer load instruction.

24. (Original) The method of claim 22 wherein:
the type of the transition is an initial transfer to the VM; and
performing the set of operations comprises marking the VMCS as cleared when receiving a request from the VMM to initialize the VMCS, determining that the VMCS is in a cleared state, performing a plurality of validation checks on at least one of VMM state information and VM state information, storing the VMM state information to the VMCS, loading the VM state information into a processor storage, marking the VMCS as launched, and beginning to execute the VM.

25. (Original) The method of claim 24 wherein the request to initialize the VMCS is a VMCS clear instruction executed by the VMM, the VMCS clear instruction including the pointer to the VMCS as an operand.

26. (Original) The method of claim 22 wherein performing the set of operations further comprises allocating an on-processor cache storage for the VMCS upon determining that the VMCS is in a cleared state, and caching information stored in the VMCS to the on-processor cache storage during operation of the VM.

27. (Original) The method of claim 22 wherein:
the type of the transition is a subsequent transfer to the VM; and
performing the set of operations comprises determining that the VMCS is not in a cleared state, performing a plurality of validation checks on at least one of VMM state information and VM state information, storing the VMM state information to the VMCS, loading the VM state information into a processor storage, and beginning to execute the VM.
28. (Previously Presented) The method of claim 27 wherein performing the plurality of validation checks comprises:
determining whether an element of the at least one of VMM state information and VM state information contained in an on-processor cache storage has been modified; and
validating the element of the at least one of VMM state information and VM state information if the element has been modified.
29. (Original) The method of claim 27 wherein loading the VM state information into the processor storage comprises:
determining that an element of the VM state information is contained in an on-processor cache storage; and
determining that the element of the VM state information has been modified in the VMCS.
30. (Original) An apparatus comprising:
a resource use determinator to identify a predefined behavior of a virtual machine monitor (VMM) with respect to one or more virtual machines (VMs); and
a resource optimizer to utilize processor-managed resources associated with the one or more VMs based on the predefined behavior of the VMM.
31. (Original) The apparatus of claim 30 wherein the predefined behavior of the VMM is any one of a first-time invocation of a VM, a subsequent invocation of a VM, a last

invocation of a VM, and a modification of content of a virtual machine control structure (VMCS) associated with a VM.

32. (Original) The apparatus of claim 30 further comprising a processor notification module in the VMM to notify a processor of the predefined behavior of the VMM.

33. (Original) The apparatus of claim 30 further comprising a VMM behavior predictor in a processor to predict the predefined behavior of the VMM

34. (Original) The apparatus of claim 30 wherein the resource optimizer is to utilize the processor-managed resources by performing at least one of allocation of one or more processor-managed resources, de-allocation of one or more processor-managed resources, verification of data stored in one or more processor-managed resources, invalidation of data stored in one or more processor-managed resources, and loading of data into one or more processor-managed resources.

35. (Previously Presented) An apparatus comprising:

a transition type determinator to determine that a transition from a virtual machine monitor (VMM) to a virtual machine (VM) is about to occur and to determine a type of the transition, the type of the transition being based on invocation information of the VM; and
a VMM operation controller to notify a processor of the type of the transition.

36. (Original) The apparatus of claim 35 wherein the VMM operation controller is to notify the processor by executing an instruction associated with the type of the transition.

37. (Original) The apparatus of claim 35 wherein the VMM operation controller is to respond to a determination that the transition is an initial transfer to the VM by allocating a memory region for a new virtual machine control structure (VMCS) associated with the VM, requesting the processor to initialize the new VMCS, requesting the processor to activate the

Docket No. 42P15752

Application No. 10/663,163

new VMCS, and requesting the processor to set execution control information, VMM state information and VM state information in the new VMCS.

38. (Original) The apparatus of claim 37 wherein the VMM operation controller is to request the processor to activate the new VMCS by executing a VMCS pointer load instruction including a pointer to the new VMCS as an operand.

39. (Original) The apparatus of claim 37 wherein the VMM operation controller is to request the processor to initialize the new VMCS by executing a VMCS clear instruction including the pointer to the new VMCS as an operand.

40. (Original) The apparatus of claim 35 wherein the VMM operation controller is to respond to a determination that the transition is a subsequent transfer to the VM by requesting the processor to update content of a virtual machine control structure (VMCS).

41. (Previously Presented) An apparatus comprising:

a notification receiver to receive, from a virtual machine monitor (VMM), a request to perform a transition from the VMM to a virtual machine (VM), the request indicating a type of the transition, the type of the transition being based on invocation information of the VM; and

an operation performer to perform a set of operations according to the type of the transition.

42. (Original) The apparatus of claim 41 wherein the notification receiver is to receive a request to perform a transition by identifying a VMM execution of an instruction associated with the type of the transition.

43. (Original) The apparatus of claim 41 wherein the notification receiver is further to receive from the VMM a pointer to a virtual machine control structure (VMCS) associated with the VM.

44. (Original) The apparatus of claim 41 wherein the pointer to the VMCS is included as an operand of a VMCS pointer load instruction executed by the VMM.

45. (Original) The apparatus of claim 41 wherein the operation performer is to respond to a VMM request for an initial transfer to the VM by determining that the VMCS is in a cleared state, performing a plurality of validation checks on at least one of VMM state information and VM state information, storing the VMM state information to the VMCS, loading the VM state information into a processor storage, marking the VMCS as launched, and beginning to execute the VM.

46. (Original) The apparatus of claim 45 wherein the operation performer is further to allocate an on-processor cache storage for the VMCS upon determining that the VMCS is in the cleared state and to cache information stored in the VMCS to the on-processor cache storage during operation of the VM.

47. (Original) The apparatus of claim 45 wherein the operation performer is to respond to a VMM request for a subsequent transfer to the VM by determining that the VMCS is not in the cleared state, performing a plurality of validation checks on at least one of VMM state information and VM state information, storing the VMM state information to the VMCS, loading the VM state information into a processor storage, and beginning to execute the VM.

48. (Original) The apparatus of claim 47 wherein the operation performer is to perform the plurality of validation checks on state information by determining whether an element of the at least one of VMM state information and VM state state information contained in an on-processor cache storage has been modified, and validating the element of the at least one of VMM state information and VM state state information if the element of the at least one of VMM state information and VM state state information has been modified.

49. (Original) The apparatus of claim 48 wherein the operation performer is to load VM state information into the processor storage by determining that an element of the VM state

information is contained in the on-processor cache storage, and determining that the element of the VM state information has been modified in the VMCS.

50. (Original) A system comprising:

a memory; and

a processor coupled to the memory; and

processor-managed resources coupled to the processor that are associated with one or more virtual machines (VMs),

wherein the processor is to identify a predefined behavior of a virtual machine monitor (VMM) with respect to the one or more VMs and to utilize the processor-managed resources based on the predefined behavior of the VMM.

51. (Original) The system of claim 50 wherein the predefined behavior of the VMM is any one of a first-time invocation of a VM, a subsequent invocation of a VM, a last invocation of a VM, and a modification of content of a virtual machine control structure (VMCS) associated with a VM.

52. (Original) The system of claim 51 wherein the processor is to identify a predefined behavior of a VMM by receiving an indication of the predefined behavior from the VMM.

53. (Previously Presented) A system comprising:

a memory to store guest software; and

a processor, coupled to the memory to receive, from a virtual machine monitor (VMM), a request to perform a transition from the VMM to the guest software, the request indicating a type of the transition, the type of the transition being based on invocation information of the guest software, and to perform a set of operations according to the type of the transition.

54. (Original) The system of claim 53 wherein the processor is to receive the request to perform the transition by identifying a VMM execution of an instruction associated with the type of the transition.

55. (Original) The system of claim 53 wherein the type of the transition is any one of an initial transfer to the VM and a subsequent transfer to the VM.

56. (Currently Amended) A machine-readable storage medium containing instructions which, when executed by a processing system, cause the processing system to perform a method, the method comprising:

determining that a transition from a virtual machine monitor (VMM) to a virtual machine (VM) is about to occur;

determining a type of the transition, the type of the transition being based on invocation information of the VM; and

notifying a processor of the type of the transition.

57. (Currently Amended) The machine-readable storage medium of claim 56 wherein notifying the processor comprises executing an instruction associated with the type of the transition.

58. (Currently Amended) The machine-readable storage medium of claim 56 wherein the type of the transition is any one of an initial transfer to the VM and a subsequent transfer to the VM.